C H A P T E R 6

Mobile PC 2001

INDUSTRY EARLY REVIEW DRAFT V.0.3 — 07/21/1999 9:39 AM–

NOTE to REVIEWERS: This is a very early draft version, and no effort has been made to reconcile changes in cross references to other chapters in the guide. Please look for comments such as this in the draft, which encourage your feedback on specific issues.

Please submit comments using the form on http://www.pcdesguide.org or by sending e-mail to comments@pcdesguide.org.

IMPORTANT: The requirements defined in this guide provide guidelines for designing PC systems that will result in an optimal user experience with typical Windows-based applications running under either the Microsoft Windows 98 "Millennium" or later or Windows 2000 Professional or later operating systems. These design guidelines are not the basic system requirements for running any version of Windows operating systems.

Contents

This chapter provides a summary of the key PC 2001 requirements for mobile PCs, docking stations, and mini-docks. Mobile PC systems have thermal, portability, battery run-time and battery life, size, weight, and connectivity tradeoffs required for their design that differ from the tradeoffs made for stationary systems.

Unless a specific requirement or exception is defined in this chapter, all requirements apply for mobile PCs as defined elsewhere in this guide. If there is a conflict with requirements or recommendations made elsewhere in this guide, the items in this chapter have precedence for mobile PCs.

Mobile PC System Design Requirements

This section summarizes the additional design exceptions and design requirements for mobile PCs.

Unless an explicit exception is stated in this section, the PC 2001 requirements apply for mobile PCs as defined elsewhere in this guide. However, if there is any conflict with requirements or recommendations stated elsewhere in this guide, the items in this section have precedence for mobile PCs.

6.1 Mobile PC performance meets Mobile PC 2001 minimum requirements

For mobile PC systems based on Intel Architecture compatible processors, minimum PC 2001 performance requirements are the following:

?? ~~233~~300 MHz processor with 128K L2 cache.

Note: The processor requirement does not specify a particular processor form factor or package type.

Note to Reviewers: Please comment on how this (and other requirements in this chapter) will apply for Mini-notebook

?? ~~32~~64 MB minimum system memory. The basic PC 2001 limitations for memory available to the operating system apply for mobile PCs, as defined in the requirement X.Y, “System memory meets PC 2001 minimum requirements.”

~~*Mini-notebook Note*~~

~~For mini-notebook systems, the minimum system performance requirement is 233-MHz processor with no L2 cache and 16-MB RAM.~~

6.2 Mobile PC supports Smart Battery or ACPI Control Method battery

~~Recommended:~~OEMs are encouraged to support Smart Battery.

?? **6.2.1 Smart Battery meets PC 2001 requirements.** If Smart Battery is implemented, the following requirements apply:

- ?? An ACPI embedded controller-based (EC) System Management Bus (SMBus) interface is required, as described in Section 13 of the ACPI 1.0a specification (or later).
- ?? The battery must support the complete command set and meet the accuracy requirements defined in *Smart Battery System Specification, Version 1.0.*
- ?? A Smart Battery Charger, if used, must comply with the command requirements defined in *Smart Battery Charger Specification, Version 1.0.*
- ?? A single-battery system that does not use a Smart Battery Charger must report the presence or absence of AC power and issue AC state change

notifications by way of the EC interface, using Smart Battery Charger commands.

?? A multiple-battery system that does not use a Smart Battery Charger must report the presence or absence of AC and issue AC state change notifications using the EC by emulating a Smart Battery Selector status register.

?? A multiple-battery system that does not use a Smart Battery Charger must report the presence or absence of AC and issue AC state change notifications using the EC by emulating a Smart Battery Selector status register.

?? If a multiple-battery system is implemented, the system can use a Smart Battery Selector that complies with the *Smart Battery Selector Specification, Version 1.0.* or the Smart Battery System Manager, Version 1.0. The battery selection or alternate control scheme that is implemented must comply with the intent of the *Smart Battery* ~~*Selector*~~*System Manager Specification.* It must expose emulated Smart Battery ~~Selector~~*System Manager* registers to the operating system.

The intent is that battery systems returning "Smart Battery System" data by way of the EC SMBus interface do so in a manner consistent with the Smart Battery System specifications. They must return all battery data, the charger status ~~register~~, and all selector or system manager registers in a manner transparent to the operating system, allowing the standard Smart Battery System drivers provided with the operating system to work properly.

All Smart Battery specifications are available at http://www.sbs-forum.org.

Note to Reviewers: Please tell us the potential impact of making the emulation items (above) a requirement.

?? **6.2.2 ACPI Control Method Battery meets PC 2001 requirements.** If an ACPI Control Method Battery, as defined in Section 11 of the ACPI 1.0a specification (or later), is implemented, the following requirements apply:

?? All data returned must be meaningful and accurate. If the accuracy cannot be guaranteed, return the unknown value, which typically is 0xFFFFFFFF.

?? If a multiple-battery system is implemented, it must follow the guidelines defined in the *ACPI Implementers Guide*, which is available from the web site at http://www.microsoft.com/hwdev/onnow/ACPIdock.htm.

?? Although most of the data fields returned by a Control Method battery are optional or recommended in the ACPI 1.0a specification (or later), the following data fields are required for PC 2001:

Field	Requirements
_BIF field	Power Unit must report 0x0000000. That is, batteries are required to report their Battery Remaining Capacity in

	mWH and their Battery Present Rate in mW.
	Design Capacity, reported in mWH.
	Design Capacity of Low.
_BST	Battery Remaining Capacity, reported in mWH.
	Battery Present Rate, reported in mW.

Recommended: _BTP support.

?? Batteries must be able to supply at least the capacity they report (Battery Remaining Capacity) at all times, even when the system is running on AC power. It is acceptable to supply more energy than they report, but they must never over-report capacity.

?? Batteries must accurately report the Battery Present Rate, providing data that is not stale.

~~6.3 [DELETE] Expansion capabilities of mobile PC are accessible to users~~

Note to Reviewers: This is not a PC 2001 requirement

6.4 Mobile PC connections use icons plus keyed or shrouded connectors

This requirement is the same as for PC 2001 desktop systems, except that for Mobile PC designs, with small-height considerations, connector icons might not fit on the back of the case. In such cases, it is acceptable to wrap the icons to the bottom of the unit or place them on the inside of an access door.

Mobile PCs are not required to implement color coding for connectors.

6.5 [REDUNDANT] Mobile PC includes one USB port

Note to Reviewers: Mobile PC 2001 implementation requirements for USB are defined in Chapter X, "Buses and Connectors."

6.6 [REDUNDANT] USB-connected device does not maintain fully on power state

Note to Reviewers: Mobile implementation details for buses will be placed in the buses chapter.

Mobile PC 2001 implementation requirements for USB are defined in Chapter X, "Buses and Connectors."

~~6.7 [DELETE] Mobile PC includes an IEEE 1394 port~~

Note to Reviewers: PC 2001 does not include recommendations

Mobile implementation details for expansion will be placed in the appropriate chapter.

6.8 Mobile PC includes CardBus

At least one 32-bit Type-2 CardBus slot (not 16-bit) is required. All CardBus implementations must comply with the requirements defined in Chapter 9, "Buses and Interfaces," including information about the default initialization of the CardBus controller under both Windows 98 and Windows 2000 Professional operating systems.

Note: Each device in a multifunction add-on device—such as a CardBus card—must separately meet the power management device class specifications for its device class and be independently power managed. This means that both device A and device B on the same add-on card do not have to be idle before the devices can be power managed.

6.9 Mobile PC keyboard and pointing device meet PC 2001 requirements

A PC 2001 Mobile PC system must have an attached pointing device, attached input devices (keyboard or speech I/O), and an attached display or heads-up display.

The internal keyboard and any built-in pointing devices, such a mouse, stylus, pen, touch pad, touch screen, trackball, stick, and so on, required for a mobile PC should use standard system-board devices. The USB port can be used to support the requirement for external pointing device and keyboard connections.

For more information, see Chapter X, "Input Devices," which also provides information about implementing the recommended Windows and Application logo keys on mobile PCs.

The BIOS and the driver for the internal pointing device must accommodate standard external pointing devices such that all features of the external device are available when it is attached to the system.

~~At a minimum, if the internal pointing device is a PS/2-type device, the BIOS must provide an option to detect when an external PS/2-type pointing device is connected at startup and disable the internal pointing device. In this case, the driver for the internal pointing device must not load. This should be the default BIOS option.~~

The *required* default BIOS option is to provide an option to disable the internal pointing device when any external~~PS/2-type~~ pointing device is detected at startup. In this case, the driver for the internal pointing device must not load.

~~6.10 [DELETE] Mobile PC includes IR devices compliant with IrDA specifications~~

Note to Reviewers: Recommendations are not include in PC 2001

6.11 Mobile PC includes support for installing the operating system

The Mobile PC system, as purchased, might not include all peripherals required for operating system installation.This basic PC 2001 requirement is met as long as it is possible for the user to obtain the required device support for operating system installation, even if it requires a separate purchase.

6.12 [REDUNDANT] Mobile PC includes audio that meets Mobile PC 2001 audio requirements

Note to Reviewers: Guidelines for Mobile PC 2001 audio are defined in the Audio chapter

6.13 ~~Mobile PC includes communications device~~Communications capabilities meet Mobile PC 2001 guidelines

As shipped, a Mobile PC can be used to access the Internet. The presence of a CardBus slot on the mobile PC meets the PC 2001 requirement for providing either a modem or network adapter, with the following exceptions:

?? If modem capabilities are integrated in the base platform, then V.34 or higher is required. All other requirements for modems must be met as defined in Chapter [X], “Modems.”

?? For a network adapter, support is optional, rather than required, for remote new system setup capabilities. All other requirements for network communications must be met as defined in Chapter [X], “Network Communications.”

Support for remote wake-up is not required to be built into Mobile PCs. ~~However, CardBus implementations that support the power management event (PME) signal meets this capability. For information about PME signal definition, see *PCI Bus*~~

~~*Power Management Interface Specification for PCI to CardBus Bridge, Revision 1.0* or later.~~

~~**6.14 [DELETE] Mobile system supports hot-pluggable devices and alternative network connections**~~

Note to Reviewers: Recommendations are not include in PC 2001

~~**6.15 [DELETE]Mobile system meets Mobile Power Guidelines '99**~~

Note to Reviewers: Recommendations are not include in PC 2001

6.16 CD or DVD ~~driveMobile system includes~~drive meets Mobile PC 2001 guidelines

If a CD drive is included in a mobile PC system or is designed as an add-on device to be attached to a mobile PC system, the minimum CD drive media transfer rate must be no less than 600 KB per second when running in the fully on (D0) state.

~~The suggested minimum CD drive media transfer rate for read operations should be 1200 KB per second or greater when running in the D0 state.~~

If a DVD drive is included in a mobile PC system or designed as an add-on device to be attached to a mobile PC system, there is no requirement for a minimum transfer rate. It is recommended that the device provide no less than 2 MB per second sustained rate anywhere on the disk media for read operations.

Note to Reviewers: Mobile implementation details will be moved to the appropriate storage chapter.

6.17 If Windows 2000 is preinstalled, mobile system meets Manageability Baseline requirements

Mobile systems that come preinstalled with Windows 2000 must comply with the following PC 2001 Manageability Baseline requirements, as defined in Chapter [X], "PC 2001 Basic Requirements":

?? 3.51, "System supports WHIIG"

?? 3.52, "System includes driver support for WMI"

?? 3.53, "Management information service provider enabled by default"

Note to Reviewers: Please comment on the need for manageability and the impact to your product roadmaps.

6.8-6.25 [REDUNDANT]

Docking System Requirements

Mobile PC docking systems allow docking of a PC, ~~with additional hardware capabilities.~~to enhance the capabilities of the mobile PC, or to facilitate its use in a

fixed work environment, such as an office or home. A docking station allows the end user to add other devices to the mobile PC system—for example, sound, network adapter, hard disks, CD drive, different display adapter, SCSI, modems, and so on.

Docking systems can support hot, warm, or cold docking. Warm docking refers to docking and undocking the mobile PC while the system is in a low power state (as defined in the ACPI 1.0 specification) but is not powered off. Hot docking refers to docking and undocking the mobile PC while the system is operating at full power and is in an active working state. Cold docking refers to docking and undocking that is done with both the PC and the docking system in an unpowered state.

Resource conflicts can occur when a mobile PC is paired with a docking ~~station~~system that allows users to add non-proprietary expansion devices to the system.

For a mobile PC/docking air, the system designer must ensure that the docking system is capable of arbitrating resources for conflicts that might occur if an expansion card is added to the docking station. However, the system designer does not need to add to the Mobile PC unit all of the PC 2001 resource-arbitration capabilities.

The requirements in this section apply for mobile designs that include a docking ~~station.~~system. There is no requirement that a mobile PC must have a docking ~~station.~~system.

Docking Definitions

This section defines the types of docking modules that interface to a mobile PC platform.

Because Port Replicators provide simple wire “pass-through” to external devices, such docks do not impose any dock-specific requirements on a PC’s operating system. Port Replicators are not “software-visible”. Docking stations (i.e. Mini-Docks or Full Docks) *are* software-visible, and they *do* need OS support to configure and control their internal active-electronics hardware resources.

?? **Mini-Dock:** A mini-dock provides external cable connections as an extension of the connector receptacles on the mobile PC unit. A mini-dock also incorporates some form of active electronics to create extended mobile PC platform features and functions. ~~The added active electronics might provide additional user-accessible CardBus slots, communication receptacles, or both, such as RS-232, IEEE 1284, IEEE 1394, and so on. The mini-dock does not provide user-accessible PCI slots, but might provide internal PCI expansion capabilities accessible only to the OEM.~~

~~A mini-dock does not have internal user-upgradeable capabilities for adding desktop peripherals or I/O expansion cards. Hence, a mini-dock can be~~

~~considered a "sealed" docking station, where all expansion capabilities are provided using external expansion ports, so that the operating system always knows what to expect about available devices. However, this does not preclude designs that include internal components that can be upgraded by the OEM or trained service personnel.~~

?? **Full Dock:** A full dock, when interconnected with the mobile PC platform, is typically designed to extend the features and functions of the mobile PC to be equivalent to that of a desktop platform system. ~~Requirements and specifications for features and functions available when a mobile PC platform has been interconnected with a full dock are, typically, the same asthose for a desktop platform system.~~

A full dock incorporates native bus expansion slots. ~~It is user-expandable to include desktop peripherals and expansion cards. A full dock is typically larger than a port replicator or a mini-dock.~~

General Docking Requirements

Note to Reviewers: The authors will add additional docking requirements based on Windows 2000 capabilities in v. 0.5 of this draft.

~~The methods for the following dock identification scenarios can be supported by the system BIOS, which requires a mobile system to be aware of each type of docking station and features it supports, or by the docking station itself, which could contain the ACPI table needed to differentiate the model, unique ID, and features in the dock. Either method would allow the system BIOS to pass this information to the operating system without actually having to support every conceivable combination.~~

All docks are required to expose their components to the operating system using ACPI tables.

Windows 2000 requires that drivers for devices in a dock must fully support dynamic loading and unloading, as well as all Windows 2000-based power management and Plug and Play messages. In certain designs, some devices that are normally considered system devices can be treated as static devices.

In the case of a desktop system, static devices might not necessarily have to have their driver be capable of dynamically unloading, for example, a custom keyboard driver or custom storage driver. However, in some docking designs, such devices are sometimes "mirrored" in a docking station. Under these conditions, the driver must be able to be unloaded dynamically; otherwise, the operating system cannot stop the device, preventing a mobile ejection.

6.26 Docking station supports PCI docking through a bridge connector

~~The system should~~If PCI docking is implemented, the system must support docking through a bridge connector, with the actual bridge on the docking station

and not on the mobile unit. The bridge can ~~be~~use positive or subtractive decoding. The bridge should create a new bus number for the docking-station resident expansion bus, assuring that devices behind the bridge are not on the same bus number as other devices in the system.

After a warm dock, the BIOS should not configure the bridge or any other devices in the docking station. Configuring the docking station devices is the responsibility of the operating system.

Notice that implementing delayed transactions for PCI-to-PCI docking bridges is required in ~~PCI 2.1~~PCI 2.2 or later only when certain timing conditions are not met. For PC 2001 design requirements,~~this is interpreted to mean that~~ delayed transactions are required only when "targets cannot complete the initial data phase within the requirements of this specification," as stated in ~~PCI 2.1~~PCI 2.2 or later. Delayed transactions, which provide a performance advantage, are a hardware-related timing issue; they are not related to operating system requirements.

6.27 Docked mobile PC supports state change notification using ACPI

When a mobile PC is "docked" to a~~mini-dock or~~ docking station, specific notification must be made using ACPI methods to enable the operating system to properly change states or enumerate new devices that appear in the system. This notification must occur during a "hot" docking event or when the system returns from a warm or cold dock.

All notification events and docking control must be implemented as defined in Sections 5.6.3 and 6.3 of the ACPI 1.0 specification.

6.28 Docked mobile PC has the ability to identify the specific model of the dock

The system must be capable of uniquely identifying to the operating system a specific system configuration. Each separate system in the same model line ~~should~~must have a unique ID. This is to prevent the problems with current implementations that require the operating system to "cycle" different docking profiles at every docking event to try to identify what specific model of dock is attached.

6.29 Docked mobile PC has the ability to uniquely identify the dock

The system must be capable of uniquely identifying an individual dock. This allows support for users that dock laptops into differently configured docks to have different features or settings at different locations and again prevents the operating system from unnecessary enumeration of the system on docking events.

6.30 Mobile PC/docking station combination meets PC 2001 requirements

There is no requirement that a mobile PC must ~~have a docking station.~~ be shipped with a docking station. If a mobile PC is shipped with a docking station, the combination must meet PC2001 requirements.

~~However, if a mobile PC supports a docking station, manufacturers must submit the combined docking station and mobile PC for PC 2001 compatibility testing, and this combination must pass testing.~~

The docking unit must be able to power the mobile system and charge the mobile system's battery under the control of the mobile system.

Some PC 2001 requirements might apply to a mobile PC/docking station combination that do not apply to the mobile PC as a standalone unit. The intent for PC 2001 is that such requirements apply only because of facilities present in the docking station. For example, if a docking station provides graphics capabilities that substitute for the graphics capabilities of the mobile unit, the PC 2001 graphics requirements apply for the mobile PC/docking station combination when the substituted graphics component is in use. If the mobile PC is supplying all graphics capabilities, ~~then~~ Mobile PC 2001 graphics requirements still apply.

This does not require that all new PC 2001 mobiles that have docking station support automatically have new docking station designs designed to meet PC 2001~~requirements. PC 2001 mobile PCs can support docking stations that have already been tested to meet earlier design guideline requirements. The combination of a PC 2001 mobile and an earlier design docking station must still be submitted for testing, and general system requirements still apply.~~requirements The relevant requirements in this case are the following:

?? The user cannot experience resource conflicts.

?? All drivers for earlier docking stations must be updated as necessary to support the pre-installed operating system.

For example, in order for older docking stations to work properly with a PC 2001 mobile PC running Windows 2000, all drivers must be updated to support dynamic loading, Plug and Play, and power management messages. This does not imply that new features must be added, but rather that the mobile system/ operating system combination must have full control over the features in the docking station.

This exception does not imply that a new docking station can comply with a reduced set of PC 2001 requirements based on an earlier design guideline. If a docking station is a new design released during the time that this design guide is in effect, such combinations of mobile and docking station must meet all PC 2001 requirements.

6.31 Docking station meets all PC 2001 system requirements

All basic PC 2001 requirements must be met by the dock and its devices, as defined in Chapter [X], "PC 2001 Basic Requirements." These include requirements for ACPI, Plug and Play, power management, and bus and device specifications.

The docking station must meet the PC 2001 BIOS requirement for multiple adapters and multiple monitors, which allows for the graphics capabilities in the mobile unit to be fully operational (either the LCD panel or external connector) in the event that a user adds another graphics adapter to the docking station.

~~Many docking stations support VCR-style docking in which the notebook is closed when docked, so the user is prevented from accessing the notebook display. It is recommended that users not be precluded from accessing their notebook display when docked and that users have the option of simultaneously using the main display on the docking station and the notebook display.~~

Windows 2000 is designed such that all devices on a docking station (whether built in or added on) must be Plug and Play devices, either based on ACPI or a bus standard described in the PC 2001 guidelines.

Note: ISA slots are not allowed in docking stations, as defined in requirement 3.28, "System does not include ISA expansion devices or slots."

6.32 Mobile/docking station interface is supported using ACPI-defined mechanisms

The mobile unit must provide docking notification using mechanisms defined in the ACPI ~~1.0 specification~~1.0a implementor's guide (or later version). Non-Plug and Play devices must be enumerated, configured, and disabled using ACPI-based methods. All notification events and docking control must be implemented as defined in Sections 5 and 6 of the ACPI 1.0a ~~specification (or later).~~implementation guide (or later version).

6.33 Mobile PC/docking station combination supports automatic resource assignment and dynamic disable capabilities

The mobile PC unit that is part of a docking system does not require all of the resource-arbitration capabilities required for expandable PC systems. However, the system as a whole must be capable of completely and dynamically disabling static onboard and add-on devices, and of freeing all the resources used by that device when the mobile unit is docked.

This requirement applies for all Plug and Play devices, but excludes fixed-resource devices such as the DMA controller, interrupt controller, and so on, as summarized in requirement 3.12, "Each bus and device meets Plug and Play specifications."

With this capability, individual devices in the mobile PC can be disabled when the unit is docked, allowing the appropriate devices in the docking station to be enabled. The system could fail if an add-on card requires resources that conflict with a device on either the mobile PC or the docking station. The mobile PC/docking station combination must be able to resolve resource conflicts among all the devices in the docking system.

This means that docking station devices must be available to replace disabled devices in the mobile PC, and these devices must meet the basic Plug and Play resource arbitration requirements for PC 2001, as described in the "PC 2001 General Device Requirements" section in Chapter [X], "PC 2001 Basic Requirements." However, it is up to the design engineer of a mobile PC/docking station combination to determine which component (mobile PC or docking station) will resolve the conflict when the mobile unit is docked.

For more information about resource arbitration when two devices such as two keyboards or two mice are present, see requirement 13.49, "Dynamic resource configuration is supported for all devices."

Note: Under Windows 2000, drive letter assignments ~~will~~do not change when drives are added or removed by way of a docking event. That is, all drives in the mobile PC~~will~~ retain their originally assigned drive letters. Designers should note this differing capability in comparison with Windows 95/98.

6.34 Docking station supports warm docking

~~Recommended: Support hot docking.~~

Docking or undocking a mobile unit from a docking station must not require powering off the system and must not require a system reboot.

Removable ATA devices in the docking station and in the mobile unit are required to report changes using ACPI-based methods.

6.35 Docking system supports fail-safe docking

The system must provide a fail-safe mechanism for attaching and detaching the mobile unit. The mechanism, in combination with operating system capabilities and methods defined in Sections 6 and 5 of the ACPI 1.0a specification (or later), must ensure the following:

?? The undock button signals the user's intent to the system.

?? The user can initiate undocking through Windows-based software choices. Notice, however, that a hardware "button" must also be provided, because experience shows that users often do not find the software option and remove mobile units without operating system notification.

?? The undock button or software choice sends a signal to the operating system so that the user is warned if resources are in danger of being lost.

?? A safe-undock indicator is provided so the user can identify when it is safe to remove the mobile unit. This can be an LED or any other mechanism chosen by the vendor. If a physical mechanism automatically undocks the mobile ~~or if hot docking is supported, then~~PC, the safe-undock indicator is not required.

There is no requirement for mechanical lockout to block the user from removing the mobile unit without operating-system notification.

~~6.36 [DELETE] Docking station includes an IEEE 1394 port~~

Note to Reviewers: Recommendations are not include in PC 2001

6.37 If audio is implemented, docking station/mobile pair meets PC 2001 audio requirements

If audio is implemented, the docking station/mobile PC pair must meet the requirements for PC 2001 audio as defined in Chapter 11, “Audio,” with additional requirements as follows:

?? The user must be able to select speakers in the mobile unit or the docking station.

~~System vendors can choose to automate the process either in the docking station or the mobile PC to meet this requirement. For example, instead of offering a UI where the user can select speakers, the system manufacturer can configure the pair to automatically turn on the docking station speakers and turn off the mobile PC speakers when in the docked configuration.~~

The ~~objective of this requirement is to ensure that users can access the highest quality audio output in any given configuration. If speakers are selected automatically, the vendor should prevent multiple outputs from occurring simultaneously. If speakers are not automatically selected, then a manual selection process must be offered to the user. Additionally, the~~ speakers should be switched off if the headphone or line-out jacks are used.

?? The docking station is not required to implement full desktop audio capabilities, but it can supplement the audio capabilities of the mobile unit.

Mini-Dock Requirements

The requirements in this section apply for any mini-dock designed for a PC 2001 mobile PC. There is no requirement that a mobile PC must have a mini-dock.

A mobile PC with a mini-dock does not need to meet the expansion card requirements and does not need to meet all the resource requirements of a mobile PC/full dock combination. ~~A mini-dock is not required to provide an undock or eject button.~~

However, some mobile PC system designs include a mini-dock that has dedicated features for networking, additional CardBus slots, a CD drive, and so on. This means that the system could have additional resource requirements to the point

that all available IRQs in the system are already allocated; in this case, the CardBus slots (for example) would not have any IRQs available, rendering them useless.

In such cases, the mini-dock must contain devices that replace any devices in the mobile PC that do not meet the IRQ, DMA, I/O port, and memory requirements for PC 2001. This allows the operating system to disable the device on the mobile PC, to enable the corresponding device on the mini-dock, and then to arbitrate resources among the remaining devices in the mobile unit and on the mini-dock.

6.38 Mini-dock supports automatic resource assignment and dynamic disable capabilities for replacement devices

A mini-dock that can accept expansion cards must contain devices that replace any devices in the mobile PC not meeting PC 2001 requirements for IRQ, DMA, I/O port, and memory resources. This allows the operating system to disable the device on the mobile PC, enable the corresponding device on the mini-dock, and arbitrate resources among the remaining devices in the mobile unit and on the mini-dock.

Devices in the system must be capable of being dynamically disabled so that the user can choose to free resources in order to allow other devices in the system to function.

Tip: To avoid resource shortages, the system designer can take advantage of the capability of Yenta compliant CardBus controllers' capability to assign a shared PCI interrupt for R2 PC Cards, rather than using IRQs, as defined in the *PC Card Design Guidelines for Windows.* For information, see the related article at http://www.microsoft.com/hwdev/cardbus/.

6.39 Mini-dock supports warm docking

Docking or undocking a mobile unit from a mini-dock must not require powering off the system and must not require a system reboot.

Removable ATA devices in the mini-dock and the mobile unit are required to report changes using ACPI-based methods.

6.40 Mini-dock supports fail-safe docking

The system must provide a fail-safe mechanism for attaching and detaching the mobile unit. The mechanism, in combination with operating system capabilities and methods defined in Sections 5 and 6 of the ACPI 1.0a specification (or later), must ensure the following:

?? The mini-dock has an undock button that signals the user's intent to the system.

?? The user can initiate undocking through a Windows-based software choice or the hardware undock button. Either choice must cause a signal to be sent to

the operating system so that the user is warned if resources are in danger of being lost.

?? A safe-undock indicator must be provided so the user can identify when it is safe to remove the mobile unit. The indicator can be an LED or any other mechanism chosen by the vendor. If a physical mechanism automatically undocks the mobile PC or if hot docking is supported, then the safe-undock indicator is not required.

There is no requirement for mechanical lockout to block the user from removing the mobile unit without operating system notification.

~~**6.41 [Delete] Mini-dock includes an IEEE 1394 port [cf. 6.36]**~~

Note to Reviewers: Recommendations are not include in PC 2001

~~**6.42DELETE**~~

PC Card Socket Controller Requirements

This section summarizes requirements and standards for socket controllers.

Note to Reviewers: We are considering moving this section out of the document to a repository for well known industry standards that don't need to be repeated in this guide.

[12.2] Controller supports industry-standard ExCA register set

The built-in software supporting 16-bit PC Card cards in Windows includes drivers for the industry-standard Exchangeable Card Architecture-compatible (ExCA-compatible) socket controllers. To be compatible with these drivers, socket-controller implementations must support the industry-standard ExCA base register set.

Notice that some controllers do not fully implement the register set and therefore are incompatible. Also, some controllers implement extended registers or enhancements. The built-in Windows drivers do not exploit these features, even though the controller might be compatible.

[12.3] System maintains mapping of IRQ Routing Register bits to system interrupt vectors

The system design must maintain the mapping of the PC Card controller's IRQ Routing Register bits to system interrupt vectors. This means that when an interrupt is programmed in the controller to occur on the IRQ*x* pin, the system's IRQ routing causes the interrupt controller to generate the interrupt vector for IRQ*x* and no other IRQ.

[12.4] IRQ connections can be determined by using the 0805 register

Windows uses the 0805 register on CardBus controllers to determine which ISA IRQs are connected to the controller. This register must select (drive low when the IRQ is asserted) the corresponding ISA IRQ when programmed with a value. It must deselect the IRQ (float high) when programmed at zero (0). This behavior must be achieved without requiring the operating system to program any non-standard registers.

Note to Reviewers: This guideline might be adjusted due to V7 of the PC Card Standard.

[12.5] CardBus controllers support both ISA and PCI interrupts

PC Card software dynamically configures the bridge to use ISA interrupts for 16-bit PC Card cards and to use Peripheral Component Interface (PCI) interrupts for CardBus cards. As defined in requirement 12.5, “IRQ connections can be determined by using the 0805 register,” and requirement 12.4, “System maintains mapping of IRQ Routing Register bits to system interrupt vectors,” CardBus controllers must maintain mapping of IRQ routing. Also, notice that systems implementing CardBus controllers must fully support PCI 2.2 as well as additional PCI requirements for IRQ routing as described in requirement X.XX, “Interrupt routing is supported using ACPI,” and requirement X.XX, “BIOS does not configure I/O systems to share PCI interrupts.”

To ensure that the Windows operating system can correctly assign ISA IRQs to 16-bit PC Cards, CardBus controllers that have parallel ISA IRQ mode must have all ISA IRQs pins, except IRQ 0 (timer), 1 (keyboard), 6 (floppy), 8 (CMOS), 13 (math coprocessor). ~~It is recommended that~~ System vendors using parallel ISA IRQ mode always should connect ISA IRQs 3, 4, 5, 7, 9, 10, 11, 12, 14, 15 and not cross wire them. Vendors using serialized IRQ mode only need to connect the serial IRQ pin, and the ISA IRQ information will be sent to the PCI chip set serially; the ISA IRQ information can specify any of IRQ 0–15.

[12.6] System supports industry-standard definition for CardBus bridges

Systems must support the definition in the *PC Card Standard Release 7 (or later) PC Card Host System Specification, PCI-to-CardBus Bridge Register Description* for CardBus controllers (PCI-to-CardBus bridges). This definition includes a common PCI Configuration Space header assigned the Header Type field value of 82h.

Windows supports this specification. Any controller features that are not part of this specification will not be used in standard drivers. The BIOS is responsible for any hardware initialization or setup required to make the controller comply with this specification, or with other requirements listed in this chapter.

Because CardBus host controllers are PCI bus bridges, they will be supported (enumerated and configured) by the PCI software in Windows in the same manner as other PCI bus bridges. For more information, see requirement X.X,

"System uses standard method to close BAR windows on nonsubtractive decode PCI bridges."

[12.7] BIOS initializes CardBus controller in 82365-compatible mode and supports backward compatibility

When 82365 compatabile modes are implemented, CardBus controllers are enumerated and configured in the same way as other PCI bus bridges. The PCI bus bridge support in Windows ~~98~~ is based on requirements for PCI interrupt routing and bridge-window configuration.Therefore, full compliance with the latest PCI specifications is a requirement for CardBus support. See the PCI section of Chapter 9, "Buses and Interfaces."

There are steps the BIOS can take to achieve backward compatibility with Windows. Specifically, the BIOS can initialize the CardBus controller in Intel 82365-compatible mode and report it as device "PNP0E03, Intel 82365-compatible CardBus controller." The requirements for BIOS POST time (CardBus controller ConfigSpace initialization) are as follows:

?? Command register (offset 0x04) set to 0x07 (IOSpaceEnable, MemSpaceEnable, BusMasterEnable).

?? RegisterBaseAddress (offset 0x10) set to 0. If support for other environments is needed, such as Windows 3.1 or MS-DOS®, some other value can be set.

?? All memory and I/O windows (offset 0x1c–0x38) set to 0.

?? Interrupt Line register (offset 0x3c) set to 0xff (no IRQ assigned). If support for other environments is needed, such as Windows 3.1 or MS-DOS, an assigned IRQ line can be set. Notice, however, that this register must be set to 0xff at the time that the device is disabled by the operating system, and then set into CardBus mode. More information about BIOS enumeration is presented later in this requirement.

?? Other controller-specific initialization as required to put the controller in 82365-compatible legacy mode.

This puts the CardBus controller into legacy mode where the Windows Socket Services driver can access it as an Intel PC Card Interface-Controller–compatible (PCIC-compatible) controller at an I/O address, for example, 0x3e0.

Notice that the BIOS must be at least PCI 2.2-compliant and must support the $PIR Interrupt Routing Table. The $PIR table must return the necessary PCI IRQ routing information, including the routing information for the CardBus controller. In general, if the CardBus controller is on the system board, there must be a slot routing entry for it in the table. If the CardBus controller is a PCI add-on card, there must be routing information entries for each PCI slot in the system.

During Plug and Play BIOS enumeration, the BIOS should report the CardBus controller as *pnp0e03 with a compatible ID of *pnp0e00 and the I/O resource of two ports, for example, 0x3e0–0x3e1.

For more information, see the white paper on CardBus host controllers and Windows compatibility at http://www.microsoft.com/hwdev/cardbus/.

[12.8] CardBus controllers do not share writable PCI Configuration Space bits

CardBus controllers are multifunction PCI devices, and Windows treats each function as an independent device. As such, there can be no sharing between functions of writable PCI Configuration Space bits, such as the Command register.

Notice that the 16-bit PC Card interface legacy-mode BAR (offset 44h in the Type 2 PCI header) is the only exception to this requirement. This BAR must be shared between the two functions in order to be compatible with the ExCA programming model.

[12.9] Each 16-bit PC Card memory window in CardBus controller has it own page register

For complete flexibility and support of typical configurations, CardBus controllers must support the independent location of legacy mode memory windows anywhere in the full system address space as recommended in the *PC Card Standard Release 7 (or later) PC Card Host System Specification, PCI-to-CardBus Bridge Register* specification.

Controllers that share a single page register among all 16-bit PC Card memory windows require that all 16-bit PC Card memory windows must be located within the same 16-MB block. This is often not possible with typical (16 MB) DRAM and bridge (positive-decode) configurations. The result is disabled cards.

Plug and Play Design for 16-bit PC Card Cards

This section summarizes the Plug and Play requirements for 16-bit PC Card cards.

The Windows operating system determines what type of card is plugged into the PC Card socket by examining the tuples on the card. For Plug and Play functionality, 16-bit PC Card I/O cards must support a set of required information and configuration tuples. The PCMCIA bus enumerator uses these tuples to identify the card, load the correct device driver, and indicate all possible configurations to the Plug and Play configuration manager. The operating system then dynamically assigns a valid configuration based on this information.

[12.10] Card supports required I/O card tuples

The following items must be implemented for any 16-bit PC Card I/O card that connects to a PC2001 system:

?? The 16-bit PC Card card must contain:

- ?? The device information tuple (CISTPL_DEVICE, 01h for cards capable of 5 V operation or CISTPL_DEVICE_0C, 1Ch for cards capable of 3.3 V operation).
- ?? The Level 1 (L1) version/product information tuple (CISTPL_VERS_1, 15h).
- ?? The configuration tuple (CISTPL_CONFIG, 1Ah).
- ?? The configuration table entry tuple (CISTPL_CFTABLE_ENTRY, 1Bh).

?? A 16-bit PC Card card with more than 64 MB Common Memory must contain the extended device information tuple (CISTPL_EXTDEVICE, 09h).

?? The L1 version/product information tuple must contain the product name and manufacturer name in the product information string (TPLLV1_INFO, byte 4).

?? The product name and manufacturer name in the L1 version/product information tuple must be composed only of ASCII characters greater than ASCII 20h and less than ASCII 7Fh.

Windows uses the information contained in the required and recommended tuples to create a unique device ID for the card and to assimilate configuration information for the device. Windows uses the device configuration tuples to determine the general characteristics of the card.

Required I/O Card Tuples

Tuple ID	Tuple code	Description and comments
01h	CISTPL_DEVICE	Device information (common memory). For non-memory cards, this tuple must be present, but the device type will be NULL.
15h	CISTPL_VERS_1	L1 version/product information strings: Product information, Product name, Product number, Other manufacturer information
1Ah	CISTPL_CONF	Configuration. Indicates the location of configuration registers and registers present.
1Bh	CISTPL_CE	Configuration table entry. Appropriate configuration requirements for I/O space, interrupts, memory, and so on should be specified.
20h	CISTPL_MANFID	Manufacturer ID. Card manufacturer ID code. Defines manufacturer for this card.
21h	CISTPL_FUNCID	Function ID. Provides function information about the card. Also includes system initialization information.

The device information tuple provides information about the memory devices used in the card's common memory space. The device type, size, and speed are used to configure the socket for efficient access to the card. This tuple must be present on 16-bit PC Card I/O cards, but the device type must be NULL.

The L1 version/product information tuple contains human-readable information about the product and its manufacturer. This information is intended to be displayed to the user where necessary. Windows uses the information contained in the product information string and product name string to construct the device ID for that card. It also scans through the tuple, starting at the very beginning and continuing to the end of the product name string.

The information gathered from the L1 version/product information tuple is used to construct the unique device ID. Because the optional third and fourth strings in the tuple are not used in the unique ID, devices that require unique numbers on each card can use these strings to store that information.

The configuration tuple tells the software where to locate the configuration registers that program the card's configuration, as well as which registers are present on the card.

Each configuration table entry tuple completely describes one valid configuration in which the card can operate. Each entry describes power, timing, I/O space, interrupt, and memory space requirements for the given configuration. Configuration software selects one of these configurations for the card based on the resources currently available in the system.

The manufacturer ID tuple (CISTPL_MANFID, 20h) and the function ID tuple (CISTPL_FUNCID, 21h) add extra flexibility to a PC Card that connects to the PC:

?? The manufacturer ID tuple provides unique information about the card manufacturer. This code is registered with PCMCIA. Windows uses the manufacturer ID tuple as one source for creating a 16-bit CRC used in the construction of the device ID.

?? The function ID tuple provides information about the class of device or what function the card provides, for example, memory, modem, disk, and so on. This information helps the software perform necessary installation tasks and locate compatible drivers. Although it is not required to make this determination, Windows uses the function ID tuple internally to determine what type of device is on the PC Card.

[12.11] Configuration table entry tuples listed in priority order

Configuration table entry tuples are placed in the preferred order for configuring the device. Windows processes the tuples in the order they are placed in the Card Information Structure (CIS). From these tuples, Windows creates a logical configuration in this order and prioritizes them in the same order. Notice that for multiple voltage cards, the voltage policy is to prioritize 3.3-volt configurations,

if they are supported by the system, over 5-volt configurations, regardless of the order of the configuration table entry tuples (CISTPL_CFTABLE_ENTRY).

[12.12] Card specifies maximum configuration options

Many older PC Cards specified fixed configurations in order to address compatibility with existing software. However, this is not the intended use for tuples; the configuration software should be responsible for compatibility. The tuples should be used only to describe its maximum configurability, ruling out configurations not supported by the hardware.

If fixed configurations must be provided for an operating system other than Windows, there must be one or more entries that specify the maximum configurability that the hardware can handle. An example of "maximum configurability" is to specify "any IRQ" rather than only IRQ 3 or IRQ 4.

Plug and Play Design for CardBus

This section summarizes the Plug and Play requirements for CardBus cards. CardBus was designed to gain the benefits of PCI in a PC Card format. Consistent with this goal, Windows support for CardBus places specific requirements on CardBus cards.

[12.13] Configuration Space meets Common Silicon Guidelines

The Common Silicon Guidelines are defined in Section 2.1 of the *PC Card Standard Guidelines, Volume 10.*

The standard for CardBus defines a PCI "Type-2" Configuration Space that is defined in Section 4.5 of Volume 11 (*PC Card Host System Specification*) of the *PC Card Standard*. The Type-2 CardBus-bridge PCI header structure was defined to be as similar to the Type-1 (PCI-to-PCI bridge) header as possible. Type-2 and Type-1 PCI headers differ only in that the Type-2 header allows 4-byte resolution in I/O Base and Limit registers, while the Type-1 header supports a coarser 4 Kbyte resolution for these registers.

CardBus cards include normal Type-0 PCI headers, with certain provisions. The quadword register located at 0x28 is used as a pointer to the CardBus Card Information Structure (CIS). CardBus cards must also implement certain Command and Status Register fields that are optional for PCI devices. CardBus cards must also provide a Memory Base-Address register (BAR) for every I/O BAR provided (so that I/O windows can be memory-mapped).

Section 2.1.3.4 of Volume 10 (*Guidelines*) of the *PC Card Standard* Release 7 (or later versions) details the Common Silicon Guidelines to which CardBus cards must adhere

To maintain compatibility with existing PCI system software and drivers for PC2001, Windows will support only CardBus cards whose Configuration Space

is designed to meet the Common Silicon Guidelines. This is a requirement because CardBus configuration is performed by the PCI software, which can deal with all aspects of PCI topology configuration, including bridging. Without the allocated fields, the cards cannot be fully treated as PCI devices and cannot be supported under Windows.

The required allocated fields are listed in the following table.

Required Allocated Fields

Field	Description and comments
Vendor ID	This read-only field contains a unique ID (in PCI space) for the card manufacturer. The PCI SIG allocates unique IDs.
Device ID Revision ID	These read-only fields are vendor-assigned values that uniquely identify the device (among all vendors of PCI or CardBus products).
Class Code	This read-only field is defined in PCI 2.2. It describes what type of device the card is.
Max_Lat Min_Gnt	These read-only fields specify the desired settings for Latency Timer values according to PCI 2.2. A value of 0 indicates the device has no major requirements for the settings of Latency Timers.
Interrupt Line	This register must be read-write and must not be connected to anything, just as on PCI cards. This register is used to store the current IRQ routing for the device.

[12.14] RESERVED fields comply with PCI 2.2

The CardBus specification also lists two RESERVED fields (offset 2C in the Configuration Space), which have since been defined in PCI 2.2. These fields are also required on CardBus cards for Windows compatibility.

Required RESERVED Fields

Field	Description
Subsystem ID	If different from Device ID
Subsystem Vendor ID	If different from Vendor ID

[12.15] CardBus card implements required and recommended tuples

For CardBus, Windows also requires the same set of card tuples recommended in the PC Card guidelines, as summarized in the following table.

Required CardBus Tuples

Tuple ID	Tuple code	Comments
04h	CISTPL_CONFIG_CB	—

05h	CISTPL_CFTABLE_ENTRY_CB	—
07h	CISTPL_BAR	—
13h	CISTPL_LINKTARGET	Required as first tuple in PC Card standard.
15h	CISTPL_VERS_1	—
20h	CISTPL_MANFID	—
FFh	CISTPL_END	Required as end-of-chain tuple in PC Card standard.
21h	CISTPL_FUNCID	Recommended in PC Card standard; required for PC2001.

Power Management for PC Card

This section summarizes the specific power management requirements for PC Card. Power management requirements for specific device classes are defined in the related chapters in Part [X] of this guide.

[12.16] Socket controller complies with device class power management reference specification

This applies for both 16-bit PC Card-only controllers and CardBus controllers.

The *PC Card Controller Device Class Power Management Reference Specification, Version 1.0* or later, provides class-specific definitions of the OnNow device power states (D0–D3) for these devices. The specification also covers device functionality expected in each power state and the possible wake-up event definitions for the class, for example, whether card insertion should wake the system.

[12.17] 16-bit PC Card cards implement power-related events using ReqAttn bit and #STSCHG mechanism

Any 16-bit PC Card card that is capable of signaling a wake-up event to the system, as defined in the device class power management reference specification for its class, must implement the ReqAttn bit and its associated enable bit in the Extended Status register, and must signal on the #STSCHG line.

[12.18] CardBus controllers and cards implement PCI/CardBus power management specifications

PCI-to-CardBus bridges and CardBus cards must comply with the requirements defined in *Section 3 (PCI Bus Power Management Interface Specification for PCI-to-CardBus Bridges) of the PC Card Standard, release 7* or later. This Specification describes the CardBus power-management interface hardware, as well as proper software use of these hardware mechanisms.

The CardBus card must use the CSTSCHG pin to signal wake-up events because there is no PME# pin on the CardBus interface, and the CardBus card must use PME_EN in the card's Configuration Space to enable wake-up events. Specifically, setting the PME_EN bit in the card's Configuration Space must provide the same behavior as setting both the GWAK and WKUP bits in the card's Function Event Mask register.

Device Drivers and Installation for PC Card

This section summarizes requirements for PC Card device drivers.

[12.19] No user intervention required for correctly installing devices

The user must not be required to perform any device-installation action other than to insert disks that contain drivers and other files.

[12.20] Device is immediately functional without restarting the system

The user must be able to begin using the device without having to restart the system. Device use begins either after installation is complete or whenever the device is inserted in the system.

[12.21] ZV-compatible PC Card driver uses DirectDraw LVE

ZV-compatible PC Card drivers must use software interfaces based on 32-bit DirectDraw Live Video Extensions (LVE) in order to configure the graphics controller to receive video input using the ZV port. This includes programming the graphics controller to configure the format of the video data, its location on screen, and so on. LVE is part of DirectX 3.0 and later versions.

ZV card device drivers must handle dynamic graphics state changes, such as resolution changes, color depth changes, and switching to and from full-screen MS-DOS–based applications.

[12.22] 16-bit PC Card card driver supports sharing of level-mode interrupts

CardBus systems support both 16-bit PC Card cards and CardBus cards. In this environment, interrupt sharing becomes an issue because CardBus controllers can use PCI interrupts, which are level-sensitive and sharable. To help alleviate interrupt limitations in CardBus systems, Windows operating systems can take advantage of PCI interrupt-sharing capabilities.

In cases where no ISA IRQs are available to a 16-bit PC Card-16 card in a CardBus controller, the operating system will assign a PCI interrupt to the card. Therefore, ~~it is recommended that~~ 16-bit PC Card-16 card drivers ~~are~~ should be updated to support interrupt sharing. However, it is a requirement that 16-bit PC Card-16 card drivers must "hook" the interrupt, whether it is sharable or not, before its hardware generates any interrupts.

See also requirement X.XX, “BIOS configures boot device IRQ and writes to the interrupt line register.”

Checklist for Mobile PC 2001

6.1 Mobile PC performance meets Mobile PC 2001 minimum requirements

6.2 Mobile PC supports Smart Battery or ACPI Control Method battery

6.4 Mobile PC connections use icons plus keyed or shrouded connectors

6.5 [REDUNDANT] Mobile PC includes one USB port

6.6 [REDUNDANT] USB-connected device does not maintain fully on power state

6.8 Mobile PC includes CardBus

6.9 Mobile PC keyboard and pointing device meet PC 2001 requirements

6.11 Mobile PC includes support for installing the operating system

6.12 [REDUNDANT] Mobile PC includes audio that meets Mobile PC 2001 audio requirements

6.13 Communications capabilities meet Mobile PC 2001 guidelines

6.16 CD or DVD drive meets Mobile PC 2001 guidelines

6.17 If Windows 2000 is preinstalled, mobile system meets Manageability Baseline requirements

6.8-6.25 [REDUNDANT]

6.26 Docking station supports PCI docking through a bridge connector

6.27 Docked mobile PC supports state change notification using ACPI

6.28 Docked mobile PC has the ability to identify the specific model of the dock

6.29 Docked mobile PC has the ability to uniquely identify the dock

6.30 Mobile PC/docking station combination meets PC 2001 requirements

6.31 Docking station meets all PC 2001 system requirements

6.32 Mobile/docking station interface is supported using ACPI-defined mechanisms

6.33 Mobile PC/docking station combination supports automatic resource assignment and dynamic disable capabilities

6.34 Docking station supports warm docking

6.35 Docking system supports fail-safe docking

6.37 If audio is implemented, docking station/mobile pair meets PC 2001 audio requirements

6.38 Mini-dock supports automatic resource assignment and dynamic disable capabilities for replacement devices

6.39 Mini-dock supports warm docking

6.40 Mini-dock supports fail-safe docking

[12.2] Controller supports industry-standard ExCA register set

[12.3] System maintains mapping of IRQ Routing Register bits to system interrupt vectors

[12.4] IRQ connections can be determined by using the 0805 register

[12.5] CardBus controllers support both ISA and PCI interrupts

[12.6] System supports industry-standard definition for CardBus bridges

[12.7] BIOS initializes CardBus controller in 82365-compatible mode and supports backward compatibility

[12.8] CardBus controllers do not share writable PCI Configuration Space bits

[12.9] Each 16-bit PC Card memory window in CardBus controller has it own page register

[12.10] Card supports required I/O card tuples

[12.11] Configuration table entry tuples listed in priority order

[12.12] Card specifies maximum configuration options

[12.13] Configuration Space meets Common Silicon Guidelines

[12.14] RESERVED fields comply with PCI 2.2

[12.15] CardBus card implements required and recommended tuples

[12.16] Socket controller complies with device class power management reference specification

[12.17] 16-bit PC Card cards implement power-related events using ReqAttn bit and #STSCHG mechanism

[12.18] CardBus controllers and cards implement PCI/CardBus power management specifications

[12.19] No user intervention required for correctly installing devices

[12.20] Device is immediately functional without restarting the system

[12.21] ZV-compatible PC Card driver uses DirectDraw LVE

[12.22] 16-bit PC Card card driver supports sharing of level-mode interrupts